Available online at www.sciencedirect.com
SCIENCE DIRECT®

Electronic Notes in Theoretical Computer Science

ELSEVIER

Electronic Notes in Theoretical Computer Science 142 (2006) 5–10

www.elsevier.com/locate/entcs

On the Security of Security Software

Invited Position Paper

Jan Münther[1]

n.runs GmbH
Zimmersmühlenweg 62, 61440 Oberursel, Germany

Abstract

Currently, security appears to be one of the strongest sales arguments for software vendors all over the world. No other sector of the software industry has undergone a similar wave of mergers and acquisitions recently as the producers of security software. Market analyses from all leading business consultants predict heavy growth in the field, and the annual figures of the major players such as Checkpoint or Symantec back up these statements. However, the main mechanisms of the industry still apply: Innovations have to be created and presented where demand is predicted and the pressure to come up with new solutions is at an all time high. Consequently, the products are pushed out the door as quickly as possible—and all too often before quality control has been dedicated the amount of time and effort that would have been implied by due diligence. While any customer of security software expects the product to enhance the security of their environment, it might actual pose new risks. Blind faith in these solutions can further contribute to compromising the overall security of a network where a substantial advantage had been expected.
The purpose of this article is to shed a little light on the situation as it really is.

Keywords: Security software, network security, firewalls

This short note summarizes the themes surveyed by Dr. Jan Münther in his invited address. Dr. Münther is an expert in network and information security who comes from industry, and we invited him precisely because we were seeking an industrial perspective at the workshop. We are including this summary in the Proceedings in the hope that it will provide researchers with clear and important issues to take into account when considering new problems to undertake.

The Editors

[1] Email: jan.muenther@nruns.com

doi:10.1016/j.entcs.2005.07.010

1 Firewalls

Firewalls have been around for years now. Filtering traffic has become a standard procedure, at least on the network perimeter, but with growing frequency, segmentation on the link layer is also reflected on the network layer. By their very nature, firewalls usually form neuralgic spots of a network—the points where networks interconnect.

1.1 High Availability

Due to their character, they often form a single point of failure: If the firewall fails, no traffic will be routed (or bridged). The firewall vendors have adressed this problem with a plethora of High Availability (HA) solutions. However, these are often far from fault free. One of the more popular solutions is based on multicast MAC addresses usage, so a virtual physical adress receives all the frames dedicated to the virtual IP address of a firewall cluster and then forwards these to the actual cluster nodes.

One of the classical attack methods in a switched network is the usage of spoofed ARP replies to overwrite the entries in a host's ARP cache, so all frames are first directed to the attacker instead of the original target. This method allows for sniffing in switched networks and the respective tools belong to the toolbox of every determined attacker. To be able to capture the most interesting traffic, the default gateway is usually one of the targets of the ARP spoofing attack (a.k.a. ARP Poison Routing, APR)—usually the firewall.

The aforementioned HA solution however fails to handle the "restore procedure" the attacker typically executes after the attack properly. In this, the attacker re-writes the original ARP entries through sending ARP replies with the original MAC addresses to all hosts it had previously spoofed the entries for. In the case of this firewalling software, the firewall stops routing packets for any other host that had been targeted in the attack for about an hour and a half. While the vendor has acknowledged the problem, no solution is known.

In another scenario, a Denial of Service (DoS) vulnerability of a certain firewall was found. In case of HA deployment, a malicious attacker can take out the first node with one packet, then the other with a second, resulting in a reboot and resynchronization period of some eight minutes. So much for HA.

Naturally, these weaknesses do not mean that HA deployments are counterproductive or superfluous, but they cannot be counted upon as free from flaws or mitigate poor network design.

1.2 In the Perimeter We Trust

One presumption is frequently encountered: The firewall protects the good and trustworthy internal network(s) from the evil that is the Internet. Often, firewalls are configured like semi-permeable membranes: While close to nothing can get in, almost everything passes in the opposite direction, from the inside to the outside.

However, such a protection scheme in no way keeps attackers from targeting the clients to gain a foothold in the network. Microsoft's Internet Explorer is still by far the most popular browser, despite being one of the most vulnerable pieces of software ever written, judged by the number of public vulnerabilities that have been reported. A dedicated attacker can attempt to deliver malicious active content on a web page through e.g. Cross Site Scripting (XSS) and achieve code execution on a client, then retrieve a back-connecting backdoor from any given location on the net and let the client connect back to another host the attacker controls, enabling him or her to interactively execute commands on the remote machine. Given the protection level that is often implemented in internal networks, the results can be catastrophic.

1.3 The Exotic Bonus

Another common misconception is the belief that the use of a non-ubiquituous architecture or Operating System will keep the firewall itself from being compromised or from vulnerabilities being found. While the vast majority of attackers surely concentrates on the more commonly found targets such as Win32 or Linux systems on IA32 platforms, the more dedicated network explorers surely possess the ability to also reverse engineer the software images of embedded devices based on ARM, PowerPC or other platforms—at least those that the IDA Pro Disassembler supports. While it may keep the notorious "script kiddies" out, it will not stop a highly skilled and determined person from taking a closer look at what you are running.

Quintessentially, firewalls are naturally not superfluous or pose a risk that is not worth taking—all in all, they surely do more good than harm. However, relying on them as a single security measure must be considered inadequate in today's attack scenarios. Despite the promises a lot of their vendors are willing to make, they are not perfect—like any other piece of software.

2 Intrusion Detection and Intrusion Prevention—The IPS Dilemma

The firewall market is fairly saturated today. However, the task of a firewall is protecting hosts from a remote takeover—it does not sound an alarm when this has happened despite the traffic filtering.

For this purpose, Intrusion Detection Systems (IDS) have been established on the market. These function like network burglar alarms: Either based on anomaly detection or more or less advanced pattern matching, they inform their operators of potential security breaches. IDS devices are strictly passive, oftentimes even deployed with only the RX wires of an Ethernet cable actually connected or through fully passive network taps, so they can neither be detected nor abused.

Their passivity is their biggest problem, too. A lot of IDS systems remain largely unmonitored and prove about as effective as a "Hacking Forbidden" sign on the door to the data center. Due to this, the idea for Intrusion Prevention Systems was born—systems that actively terminate the connection when a security breach has been detected, typically either through sending an appropriately shaped RST flagged TCP packet or an ICMP type 3 message of varying codes. The usual placement of these systems is behind the firewalls, directly before the servers they are supposed to protect. But, since in contrast to their passive IDS brothers they need to actively produce network traffic, they must be fully connected to the actual network the servers communicate over. This makes them an attractive target—if an attacker manages to take over the IPS system, a direct foothold inside the network behind the firewall, directly before the servers has been found.

Unfortunately, this is far from theoretical. To detect typical attacks, IPS and IDS systems have to decode traffic up to the application layer and understand the logic of the high-level protocol to trigger an alert when its specifications are violated. Everyone who has ever written a piece of software that decoded a single protocol alone knows that this can become an immensely complex task. Not too surprisingly, these "decoding engines" are sometimes heavily bug ridden, and IPS devices do not make an exception.

One vendor has recently even been struck by a worm which has abused a vulnerability in the protocol parsing engine of all their Intrusion Prevention and Intrusion Detection software to control its victims.

Given these risks, do Intrusion Prevention devices really enhance your networks security?

3 Antivirus Software

Maybe the most common type of security software are the malware detecting desktop and server products usually referred to as Antivirus (AV) Software. Even this usually rather passive component of a security infrastructure can become a risk of its own.

Some vendors are neglective with their update procedures. One even downloads a standard Win32 PE executable with a predictable name from a predictable location—and executes it without the verification of any hash value or other checks. When an attacker can fool the host into downloading the file from a different location, a different file can be executed with the rights the service runs under—which is normally Local NT Authority / System, resulting in a complete takeover. One might argue that the AV software itself would detect the backdoor. This is only partly true. A custom-written backdoor that does not match any patterns or does not trigger any heuristics engine's alert will not get reported, and it is perfectly possible to adhere the original updates the service expects to the backdoor program.

Other factors include vulnerable admistration interfaces to server-based AV software, such as web-based configuration consoles for virus scanners on mail gateways. These are potentially prone to almost every attack vector known for web applications to exist. Or antivirus software may be trusted with excessive local user rights, which in case of the Win32 desktop applications may result in the software being abused for the execution of Shatter Attacks to escalate privileges locally.

4 Virtual Pirate Networks

Virtual Private Network products have triumphantly made their way into common IT infrastructure. Typical uses include the linking of two networks through an untrusted third network (typically the Internet) or allowing roaming users ("road warriors") the connection to the internal network from every point in the world, as long as they can connect to the Internet. The two most popular protocols for these purposes are the Point to Point Tunneling Protocol (PPTP) and the IETF standardized IPSecurity protocol suite (IPSec). PPTP has been criticized due to usage of weak encryption for its authentication procedures (basically PPP and a GRE tunnel in combination, using MS-CHAPv2 for the authentication itself).

IPSec has somewhat evolved as the industry standard. However, the key exchange is executed through the ISAKMP protocol, which is immensely complex. As mentioned earlier, the parsing of complex protocols is one of the most

common sources of failures and thus vulnerabilities. The author has yet to see any IPSec based Win32 VPN software that was not prone to one or more classical buffer overflows due to flaws in the ISAKMP parsing engine.

Another problem is the level of trust that network architects tend to grant in VPNs. While the other end of the tunnel is usually not under the same control as the local network itself, fairly often the two are interconnected without any second thoughts. While the Internet is seen as a hostile environment, the roaming users or the remote network are all too often allowed to connect to the private network without further filtering or traffic inspection.

IPSec based VPNs can handle the authentication based on either certificates or a shared secret, the latter essentially being a passphrase which is used on both sides of a tunnel. The shared secret variant is often picked for site-to-site connections. These are usually the connections of higher interest for an attacker. However, the passphrase-based authentication is prone to the same brute force or dictionary attacks as any other network service, whereas a certificate in its complexity is close to impossible to guess.

While VPNs offer some great possibilities, their specific vulnerabilities and the attack vectors they create must always be taken into consideration. A badly planned and possibly worse setup VPN can turn into a security hazard instead of a surplus.

5 Conclusion

Security software is just as prone to vulnerabilities as all other software, despite keywords such as “secure”, “safe” or “anti” being used in its advertising. No software can replace decent planning and reasonable network design as well as good system administration—and definitely not mitigate generic security problems deriving from custom software.

Since particularly security software is typically used in sensitive areas of IT, certain precautions should apply. As a general rule, never rely on a single line of defense. If it fails, the entire security model is compromised, and all products can be flawed. Heterogenity can be a blessing!